



①9 BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENT- UND  
MARKENAMT

⑫ **Offenlegungsschrift**  
⑩ **DE 197 58 032 A 1**

⑤1 Int. Cl.<sup>6</sup>:  
**G 06 F 13/40**  
G 06 F 13/38  
G 08 C 15/06  
H 04 L 12/40

②1 Aktenzeichen: 197 58 032.7  
②2 Anmeldetag: 29. 12. 97  
④3 Offenlegungstag: 1. 7. 99

DE 197 58 032 A 1

⑦1 Anmelder:  
NEC Electronics (Europe) GmbH, 40472 Düsseldorf, DE  
  
⑦4 Vertreter:  
Glawe, Delfs, Moll & Partner, Patentanwälte, 80538 München

⑦2 Erfinder:  
Kemper, Rolf Wilhelm, 46485 Wesel, DE; Usling, Ralf, 41849 Wassenberg, DE; Eitze, Jens Kurt, 40545 Düsseldorf, DE

⑤6 Für die Beurteilung der Patentfähigkeit in Betracht zu ziehende Druckschriften:

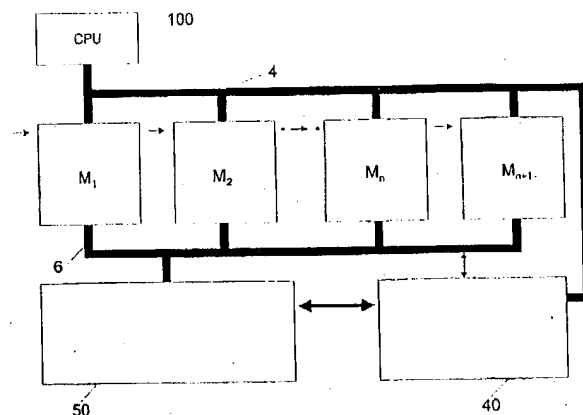
DE 41 13 100 C2  
DE 37 05 530 C2  
DE 195 34 330 A1  
DE 41 40 017 A1  
DE 39 27 968 A1  
US 53 29 618  
WO 90 16 026 A1

ETSCHBERGER, Konrad, u.a.: CAN-Netzwerke in verteilten Steuerungssystemen. In: Elektronik 26/1991, S.42-48;  
MALIVERNEY, Anita: Die nächste Generation. In: Elektronik, 12/1996, S.64-68;

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen**

⑤4 Schnittstelle für mehrere CAN-Datenverarbeitungsnetzwerke und Verfahren zum Betreiben einer Schnittstelle

⑤7 Die Erfindung betrifft eine Schnittstelle zur Koppelung von mindestens 2 CAN-Netzwerken mit jeweils verteiltem CAN-Rechnerknoten. Eine derartige Schnittstelle dient der Übertragung von Information von einem der CAN-Netzwerke an ein anderes.  
Um dabei die Belastung der zentralen Prozessoreinheit weitgehend zu minimieren und die Schnittstelle zur Anbindung weiterer Netzwerke flexibler zu gestalten, wird erfindungsgemäß vorgeschlagen, daß die Schnittstelle einen zentralen Speicher zur Speicherung von Informationen aufweist, wobei jedem der CAN-Netzwerke ein eigener Speicherbereich zugeordnet werden kann.  
Vorzugsweise sind ein eigener, zusätzlicher Maschinenbus und eine Speicherzugriffssteuerung vorgesehen, so daß die zentrale Prozessoreinheit durch die Verwaltung von Speicherzugriffen nicht zusätzlich belastet ist.



DE 197 58 032 A 1

BEST AVAILABLE COPY

Die Erfindung betrifft eine Schnittstelle (Controller) gemäß dem Oberbegriff des Patentanspruchs 1 sowie ein Verfahren zum Betreiben der Schnittstelle.

Datenverarbeitungsanlagen mit verteilten Rechnerknoten, die über einen seriellen Datenbus verbunden sind, finden überwiegend in industriellen und automobilen lokalen Netzwerken Anwendung. Die Rechnerknoten umfassen dabei Datenverarbeitungsgeräte oder Signalverarbeitungsgeräte, die für eine bestimmte Anwendung ausgelegt sind. Hierzu zählt der Datenaustausch zwischen Steuergeräten, Sensoren und Stellgliedern.

Ein Beispiel für die oben definierten Datenverarbeitungsanlagen ist das Controller Area Network (CAN), das mit einem Standardprotokoll, derzeit 2.0B arbeitet.

Jede CAN-Einheit, d. h. jeder Rechnerknoten, steht dabei über einen seriellen Datenbus mit allen anderen Rechnerknoten in Verbindung, und zum Austausch bzw. zur Verwaltung und Verarbeitung von Daten sind bei CAN im wesentlichen zwei unterschiedliche "Ebenen" definiert, nämlich eine Protokollebene und eine Objektebene. Daten werden zwischen den CAN-Einheiten in Form sogenannter "Mitteilungsobjekte" ausgetauscht. Dabei umfaßt ein "Mitteilungsobjekt" neben den eigentlichen Daten, die zu verarbeiten sind, noch beispielsweise ein Identifikationsfeld (ID), Statusfelder, Felder zur Taktsteuerung etc. Im Sinne der vorliegenden Anmeldung umfaßt der Begriff "Ebene" sowohl eine Implementierung durch Software als auch durch Hardware als auch durch Kombinationen von beiden.

Fig. 1 zeigt den schematischen Aufbau einer CAN-Einheit. Dargestellt ist die Protokollebene 1, die mit einem externen CAN-Bus 5 in Verbindung steht und verantwortlich ist für das korrekte Handhaben von Daten auf dem CAN-Bus. Insbesondere werden Mitteilungsobjekte auf korrektes Datenformat überprüft, und eingehende Mitteilungsobjekte werden von serielltem Datenformat auf paralleles Datenformat gewandelt.

Die sich daran anschließende Objektebene 2 dient der Filterung von Daten, und insbesondere der Datenverwaltung und bildet die Schnittstelle zwischen der Protokollebene und einem CAN-Speicher 3. Die Objektebene übernimmt die Datenverwaltung und die Speicherung von Daten und gibt über den Speicher 3 erforderliche Daten an eine nicht dargestellte Steuereinheit.

Der CAN-Speicher 3 ist normalerweise direkt mit der Objektebene verbunden und beispielsweise beim sogenannten "Full CAN" als RAM mit zwei Eingängen (Dual Port) ausgebildet. Die Darstellung der Fig. 1 ist lediglich eine grobe schematische Struktur. Tatsächlich können die Funktionen zwischen Objektebene und Protokollebene verschoben sein, und häufig ist eine eindeutige Trennung nicht möglich. Ein Beispiel für ein bekanntes CAN-System ist in der DE-A-41 40 017 gezeigt, auf deren Offenbarung hier ausdrücklich Bezug genommen wird.

Auf dem Gebiet der Automobiltechnik besteht zunehmend der Bedarf, eine Steuerung mit mehr als einem Netzwerk aufzubauen, beispielsweise ein Netzwerk für die Motorsteuerung und ein Netzwerk für die sogenannte "Body Electronic" wie Fensterheber, Beleuchtung, etc. Damit wird es aber auch erforderlich, entsprechende Schnittstellen zur Verfügung zu stellen, um Daten zwischen den einzelnen Netzwerken austauschen zu können, beispielsweise bei einer kombinierten Antischlupf- und ABS-Regelung. Mögliche Realisierungen einer derartigen Schnittstelle sind in den Fig. 2 und 3 dargestellt.

In Fig. 2 ist der "klassische" Weg dargestellt, um Daten zwischen unterschiedlichen Netzwerken auszutauschen. Die

in Fig. 2 dargestellte Schnittstelle 20 umfaßt jeweils zwei CAN-Einheiten 10, 10', die mit unterschiedlichen Netzwerken über ihre zugeordneten Busse 5 und 5' in Verbindung stehen. Jede der CAN-Einheiten 10, 10' entspricht im wesentlichen vom Aufbau her der in Fig. 1 dargestellten Einheit 10. Die Einheiten sind miteinander über einen internen Bus 4, verbunden, und zusätzlich ist eine zentrale Prozessoreinheit (CPU) 100 vorgesehen. Die Daten werden durch die CPU von der ersten CAN-Einheit 10 gelesen und dann in die andere Einheit 10' eingeschrieben. Dies führt dazu, daß die Belastung der CPU abhängig ist von der Anzahl der Datenübertragung an, der Anzahl der zu übertragenden Bytes etc.

Eine alternative Ausführungsform, die kein Stand der Technik ist, zeigt die Fig. 3. Zusätzlich zu den Bereichen der Fig. 2 ist noch ein Brückenmodul 12 vorgesehen, das den Datentransfer unterstützt. Das Brückenmodul könnte als Hardwareimplementierung ausgeführt sein, und die Funktionsweise des Brückenmoduls für den direkten Datenübergang zwischen zwei CAN-Einheiten muß abhängig von der Anwendung angegeben werden. Häufig ist zusätzlich noch eine CPU-Aktivität erforderlich, da nicht alle Übertragungsmechanismen durch eine solche Brücke realisierbar sind.

Des weiteren ist zu beachten, daß zur Umsetzung der in den Fig. 2 und 3 dargestellten Ansätze vorausgesetzt wird, daß die einzelnen CAN-Einheiten 10, 10' zur Realisierung der Schnittstelle kopiert werden; d. h., das Layout der CAN-Einheit wird auf einen Chip (Die) entsprechend der Anzahl der verschiedenen Anschlüsse (Netzwerke) kopiert. Dabei werden alle Einheiten wie die Protokollebene, die Objektebene und der Mitteilungsobjektspeicher kopiert und entsprechend der Schnittstellengröße (d. h. abhängig von der Anzahl der Ein- und Ausgänge) verschaltet. Damit wird aber für jede Schnittstelle abhängig von der Anzahl der CAN-Einheiten eine eigene Konfiguration benötigt, und bei einer Änderung des Aufbaus muß das gesamte Layout geändert werden. Schließlich ist der Erweiterung der Schnittstelle durch zusätzliche CAN-Einheiten eine Grenze durch die Belastung der CPU gesetzt.

Demgegenüber liegt der Erfindung die Aufgabe zugrunde, eine Schnittstelle der Eingangs genannten Art zu entwickeln, bei der die Belastung der zentralen Prozessoreinheit weitgehend minimiert ist und die sich durch hohe Flexibilität auszeichnet; des weiteren soll ein Verfahren zum Betreiben einer derartigen Schnittstelle angegeben werden.

Diese Aufgabe wird erfindungsgemäß dadurch gelöst, daß ein zentraler Mitteilungsobjektspeicher vorgesehen ist, wobei jeder Eingabe/Ausgabe-Einheit (CAN-Einheit) ein Speicherbereich zugewiesen werden kann.

Es ist somit nicht erforderlich, für jede CAN-Einheit einen eigenen Speicher auszugestalten, der von der Größe abhängig von den CAN-Einheiten des jeweiligen Netzwerkes ausgelegt werden müßte. Abhängig vom tatsächlich erforderlichen Speicherbereich kann deshalb im zentralen Speicher genausoviel Speicherplatz zugewiesen werden, wie für die betreffende Implementation erforderlich ist. Für die Erweiterung der Schnittstelle zum Anschluß eines weiteren Datennetzwerkes muß deshalb lediglich zusätzlicher Speicherplatz ausgewiesen oder vorhandener Speicherplatz umgewidmet werden, und die Protokoll- und die Objektebene für eine Eingabe/Ausgabe-Einheit müssen erstellt werden.

Vorzugsweise ist zusätzlich zu dem internen Bus, der die CPU mit den Eingabe/Ausgabe-Einheiten verbindet, ein Maschinenbus vorgesehen, über den der Datentransfer zwischen den Eingabe/Ausgabe-Einheiten und der CPU einerseits und dem zentralen Speicher andererseits vorgenommen wird.

Vorzugsweise ist dabei eine zentrale Speicherzugriffssteuerung vorgesehen, die sämtliche Datenübertragungen

vorn und zum Speicher steuert.

Jede Eingabe/Ausgabe-Einheit umfaßt dabei vorzugsweise eine Protokollebene und eine Objektebene.

Der zentrale Mitteilungsobjektspeicher ist vorzugsweise als inhaltsadressierbarer Speicher (CAM) und/oder als Freizugriffsspeicher (RAM) ausgebildet; zusätzlich kann noch ein Ereignis-Folge-Speicher vorgesehen sein, in dem eine Folge verzeichnet ist, die ausgelöst werden soll, wenn eine Statusänderung eines Mitteilungsobjektes festgestellt wird.

Um den Datenaustausch zwischen einzelnen Eingabe/Ausgabe-Einheiten zu erleichtern, ist vorzugsweise zumindest ein Brückenmodul vorgesehen, das in gleicher Weise wie die Eingabe/Ausgabe-Einheiten zwischen den Maschinenbus und den internen Bus geschaltet ist. Ausgehend von Daten, die im Mitteilungsobjektspeicher eingespeichert sind und/oder abhängig von Statusänderungen von Mitteilungsobjekten kann das Brückenmodul Daten von einem Speicherbereich in einen anderen Speicherbereich kopieren oder dahin übertragen.

Insgesamt wird somit erfindungsgemäß ein modularer Aufbau der Schnittstelle erzielt, bei dem verschiedene Einheiten der Schnittstelle als Module ausgebildet sind, die lediglich zwischen den internen Bus und dem Maschinenbus geschaltet sind. Die Module können als CAN-Module (Eingabe/Ausgabe-Einheit) mit Protokoll- und Objektebene ausgestaltet sein oder auch als Brückenmodule zum Übermitteln von Daten zwischen den Netzwerken. Schließlich ist es auch möglich, selbstdefinierte funktionelle Module in gleicher Weise zu implementieren; durch Einsatz einer Ereignis-Folge-Steuerung kann beispielsweise die Änderung eines Mitteilungsobjektes (Ereignis) eine bestimmte Folge (Aktion) auslösen.

Jeder Speicherzugriff wird über die zentrale Speicherzugriffssteuerung vorgenommen, so daß eine vollständige Kontrolle über alle Speichervorgänge gegeben ist.

Erfindungsgemäß wird jedem Modul (Eingabe/Ausgabe-Einheit, Brückenmodul, Funktionsmodul etc.) eine eigene Maschinennummer zugeordnet, und das Mitteilungsobjekt jedes einzelnen Netzes wird durch die Maschinennummer ergänzt. Auf diese Weise ist es sehr einfach möglich, jedes einzelne Mitteilungsobjekt innerhalb der Schnittstelle dem entsprechenden Datenverarbeitungsnetzwerk (CAN-Netzwerk) zuzuordnen. Auch kann dadurch der entsprechende Speicherbereich für jedes einzelne Modul zugewiesen werden. Dies erweist sich insbesondere bei Verwendung eines inhaltsadressierbaren Speichers (CAM) als vorteilhaft, in dem zumindest ein Teil des Mitteilungsobjektes, vorzugsweise die Maschinennummer und das Identifikationsfeld, gespeichert ist; restliche Daten wie beispielsweise Semaphore, Statusfelder etc. können in einem entsprechend zugeordneten RAM gespeichert werden.

Ausführungsformen der Erfindung werden anhand der beigefügten Zeichnungen erläutert. Es zeigen:

Fig. 1 den schematischen Aufbau eines CAN-Knotens,

Fig. 2 den Aufbau einer konventionellen Schnittstelle,

Fig. 3 einen möglichen Ansatz zur Vereinfachung der Datenübertragung zwischen CAN-Einheiten,

Fig. 4 den schematischen Aufbau einer erfindungsgemäßen Schnittstelle und

Fig. 5 den schematischen Aufbau eines Mitteilungsobjektspeichers.

Gemäß Fig. 4 ist die erfindungsgemäße Schnittstelle modular aus verschiedenen Modulen  $M_1, M_2, \dots, M_n, M_{n+1}$  aufgebaut. Jedes Modul ist zwischen einen internen Bus 4 und einen Maschinenbus 6 geschaltet und kann ein CAN-Modul (mit Protokollebene und Objektebene), ein Brückenmodul zur Übertragung von Daten zwischen CAN-Modulen oder ein Modul mit frei definierbarer Funktion sein. Jedes

CAN-Modul ist dabei mit dem zugeordneten CAN-Bus (nicht dargestellt) verbunden und kommuniziert mit dem entsprechenden Netzwerk.

Mit dem internen Bus 4 ist weiterhin eine zentrale Prozeßsoreinheit 100 verbunden, die Daten von den einzelnen Modulen empfängt und an diese Daten übermittelt.

Mit dem Speicherbus ist einerseits ein zentraler Mitteilungsobjektspeicher 50 und eine Speicherzugriffssteuerung 40 verbunden. Die Speicherzugriffssteuerung steuert alle Zugriffsvorgänge auf dem zentralen Mitteilungsobjektspeicher 50.

Wie bereits erwähnt, umfaßt jedes CAN-Modul die Protokollebene und die Objektebene, jedoch keinen eigenen Speicher. Der gesamte Mitteilungsobjektspeicher ist als gemeinsamer Speicher 50 ausgestaltet, der formal unabhängig von dem einzelnen Modulen ist. Um eine Verbindung zwischen unterschiedlichen Speicherplätzen des Speichers 50 und den Modulen herzustellen, wird eine Maschinennummer dem Mitteilungsidentifikationsfeld (ID) zugefügt. Ein bestimmtes Modul erhält eine Maschinennummer, die beispielsweise durch Software zugewiesen werden kann. Auf diese Weise ist es nicht erforderlich, durch feste Hardwareverbindungen zwischen Nachrichten zu unterscheiden, die zu unterschiedlichen CAN-Netzen (CAN-Modulen) oder zu anderen Arten von Modulen (wie z. B. Brückenmodule) gehören.

Zur Erläuterung der Datenstruktur soll Bezug auf Fig. 5 genommen werden. Dort ist schematisch der Aufbau eines Speichers 50 dargestellt. Erfindungsgemäß setzt sich der Speicher 50 aus einem inhaltsadressierbaren Speicher CAM 50<sub>1</sub> und einem Freizugriffsspeicher RAM 50<sub>2</sub> zusammen.

In Fig. 5 ist schematisch eine Reihenfolge von Mitteilungsobjekten 1, 2, 3, 4, 5, ..., N dargestellt, die in dieser Folge im CAM-Speicher 50<sub>1</sub> gespeichert sind. Jedes Mitteilungsobjekt umfaßt erfindungsgemäß eine Maschinennummer MN, die das zugeordnete Modul und damit, wenn es sich um ein CAN-Modul handelt, das zugeordnete Netzwerk angibt. Desweiteren ist im CAM das Identifikationsfeld für das Mitteilungsobjekt gespeichert, das beispielsweise die Art der Nachricht angibt.

Dem CAM ist weiterhin der RAM 50<sub>2</sub> zugeordnet, in dem in derselben Reihenfolge entsprechende weitere Daten eines Mitteilungsobjektes gespeichert sind. Die Speicherplätze des CAM und des RAM sind einander zugeordnet, d. h. jeder Speicherplatz des CAM hat eine Entsprechung im RAM. Somit kann durch einfachen Zugriff auf den inhaltsadressierbaren Speicher CAM sehr einfach ein Mitteilungsobjekt aufgefunden werden, das mit einer bestimmten Maschinennummer und einem bestimmten Identifikationsfeld übereinstimmt. In dem zugeordneten Speicherbereich des RAM können dann die zugehörigen weiteren Daten des Mitteilungsobjektes aufgefunden werden. Der Grund für diese Zweiteilung des Speichers liegt darin, daß einerseits ein CAM sehr aufwendig aufgebaut ist, dafür eine sehr schnelle Adressierung zuläßt, und daß andererseits ein RAM sehr einfach aufgebaut ist, aber andererseits Suchroutinen sehr langsam ablaufen. Durch diese Kombination eines CAM und eines RAM können beide Vorteile genutzt werden.

Im RAM sind für jedes Mitteilungsobjekt weitere Datenfelder vorgesehen, beispielsweise Semaphore, Zeitmarken, Ereignisfolgen und die eigentlichen Daten.

Semaphore sind dabei Bitfolgen, die Zustände von Mitteilungsobjekten angeben (beispielsweise Aktualität, etc.). Die Zeitmarke dient der Synchronisation unterschiedlicher Datenverarbeitungsnetze, und hinsichtlich der Einzelheiten der Taktsynchronisation wird auf die Eingangs erwähnte DE-A 41 40 017 verwiesen.

Neben dem eigentlichen Datenfeld ist erfindungsgemäß noch ein Ereignis-Folge-Feld vorgesehen. Das Ereignis-

BEST AVAILABLE COPY

Folge-Feld verweist bei einer möglichen Zustandsänderung der Semaphore auf einen Speicherbereich eines Ereignis-Folge-Speichers. Beispielsweise kann in dem Speicher 50<sub>3</sub> vorgesehen sein, daß bei Auftreten eines bestimmten Ereignisses (Zustandsänderung einer Semaphore eines Mitteilungsobjektes) eine bestimmte Folge eintreten soll, z. B. Änderung eines anderen Mitteilungsobjektes oder generieren eines neuen Mitteilungsobjektes, wie beispielsweise durch die Pfeile B und A in Fig. 5 dargestellt ist.

Auf diese Weise kann auch eine Schleife von Ereignisfolgen ausgelöst werden, wie durch den Pfeil C in Fig. 5 dargestellt ist. Ändert sich beispielsweise ein Mitteilungsobjekt, so verweist die Ereignisliste im RAM 50<sub>2</sub> auf eine entsprechende Folge (Aktion) ACT1 im Ereignis-Folge-Speicher 50<sub>3</sub>; wobei die im Speicher 50<sub>3</sub> angegebene Aktion ACT1 ... ACTN wieder die Änderung eines Mitteilungsobjektes verursacht, was wieder ein Ereignis darstellt. Auf diese Weise können auch aufwendige Steuerabfolgen in einfacher Weise realisiert werden.

Wie bereits erwähnt, steuert die zentrale Speicherzugriffssteuerung 40 den Zugriff auf den Mitteilungsobjektspeicher, und zwar vorzugsweise nach einem Zugriffsplan. D.h. die Speicherzugriffssteuerung stellt jedem Modul und der CPU 100 ein Zeitfenster für den Speicherzugriff zur Verfügung.

Wie ein derartiger Zugriffsplan zu erstellen ist, ist dem Fachmann geläufig und wird deshalb hier nicht im einzelnen dargestellt. Um jedoch das erfindungsgemäße modulare Konzept konsequent umzusetzen empfiehlt es sich, die Speicherzugriffsberechtigung jeweils von einem Modul auf das nächstfolgende bzw. die CPU in einer vorgegebenen Reihenfolge oder nach Prioritäten zu übergeben. Auf diese Weise kann der Speicherzugriff auch dann in einer sehr einfachen Weise realisiert werden, wenn ein zusätzliches Modul einer bestehenden Schnittstelle zugefügt werden soll.

Erfindungsgemäß wird jeder Speicherzugriff durch die zentrale Speichersteuerung gesteuert. Kein Teil des Systems (einschließlich der CPU) hat einen direkten Zugriff auf den Mitteilungsobjektspeicher. Jede Anforderung, unabhängig davon, von welchem Modul oder von der CPU selbst, wird durch die zentrale Steuereinheit verwaltet. Dies ermöglicht, daß alle Speicheränderungen überwacht können und die Semaphore entsprechend gehandhabt werden können.

Zusammenfassend ist zu erwähnen, daß erfindungsgemäß die Bestandteile der Schnittstelle modular aufgebaut sind, und zwar aus CAN-Modulen, Brückenmodulen und zusätzlichen Funktionsmodulen, die alle in gleicher Weise zwischen den internen Bus und dem Maschinenbus geschaltet sind. Die einzelnen Module umfassen keinen eigenen Speicher, sondern ihnen ist ein Speicherplatz in einem zentralen Mitteilungsobjektspeicher zugeordnet, wobei der Zugriff auf den zentralen Speicher über eine Speicherzugriffssteuerung erfolgt.

Durch diesen modularen Aufbau ergibt sich eine erhebliche Flexibilität, und das System ist gemäß den Anforderungen eines Anwenders einfach skalierbar. Erweiterungen durch zusätzliche Module sind problemlos realisierbar. Anwenderspezifische Funktionen können über eigene selbstdefinierte Module realisiert werden oder auch über eine Ereignis-Folge-Steuerung.

Alle Steuerungen sowie Brückenmodule, selbstdefinierte Module, Ereignis-Folge-Steuerung kann über Hardware implementiert werden, so daß die Belastung der CPU deutlich reduziert ist.

#### Patentansprüche

1. Schnittstelle zur Koppelung von mindestens zwei

CAN-Netzwerken mit jeweils verteilten CAN-Rechnerknoten mit mindestens zwei Eingabe/Ausgabe-Einheiten zum Austausch von Mitteilungsobjekten mit den CAN-Netzwerken und

einem internen Bus, über den die Eingabe/Ausgabe-Einheiten miteinander in Verbindung stehen,

#### gekennzeichnet durch

einen zentralen Speicher zur Speicherung von Mitteilungsobjekten, wobei jeder Eingabe/Ausgabe-Einheit ein Speicherbereich zugeordnet werden kann.

2. Schnittstelle nach Anspruch 1, gekennzeichnet durch einen zusätzlichen Maschinenbus, der mit den einzelnen Eingabe/Ausgabe-Einheiten und dem Mitteilungsobjektspeicher verbunden ist.

3. Schnittstelle nach Anspruch 1 oder 2, gekennzeichnet durch eine zentrale Speicherzugriffssteuerung, die den Zugriff auf den zentralen Mitteilungsobjektspeicher steuert.

4. Schnittstelle nach Anspruch 1, 2 oder 3, gekennzeichnet durch eine zentrale Prozessoreinheit, die mit den einzelnen Eingabe/Ausgabe-Einheiten und dem zentralen Mitteilungsobjektspeicher verbunden ist.

5. Schnittstelle nach einem der Ansprüche 1 bis 4, dadurch gekennzeichnet, daß jede Eingabe/Ausgabeneinheit eine Protokollebene zum Prüfen des Datenformats von Mitteilungsobjekten und eine Objektebene zur Verwaltung von Mitteilungsobjekten aufweist.

6. Schnittstelle nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, daß der Mitteilungsobjektspeicher einen inhaltsadressierbaren Speicher (CAM) und/oder einen Freizugriffsspeicher (RAM) aufweist.

7. Schnittstelle nach einem der Ansprüche 1 bis 6, gekennzeichnet durch einen Ereignis-Folge-Speicher.

8. Schnittstelle nach Anspruch 7, dadurch gekennzeichnet, daß der Ereignis-Folge-Speicher Teil des zentralen Mitteilungsobjektspeichers ist.

9. Schnittstelle nach einem der Ansprüche 1 bis 8, gekennzeichnet durch zumindest ein Brückenmodul, das mit dem internen Bus und dem Maschinenbus verbunden ist und Mitteilungsobjekte zwischen verschiedenen Eingabe/Ausgabe-Einheiten austauscht.

10. Schnittstelle nach einem der Ansprüche 1 bis 9, dadurch gekennzeichnet, daß jede Eingabe/Ausgabe-Einheit, jedes Brückenmodul und gegebenenfalls selbstdefinierte Funktionseinheit als Modul aufgebaut ist, das zwischen den internen Bus und den Maschinenbus geschaltet ist.

11. Schnittstelle nach Anspruch 10, dadurch gekennzeichnet, daß der Speicherzugriff jeder Einheit, jedes Moduls oder der CPU über die zentrale Speicherzugriffssteuerung erfolgt.

12. Verfahren zum Betreiben einer Schnittstelle nach einem der Ansprüche 1 bis 11, wobei jede Eingabe/Ausgabe-Einheit mit ihrem zugehörigen CAN-Netzwerk Mitteilungsobjekte austauschen kann, dadurch gekennzeichnet, daß man jeder Eingabe/Ausgabe-Einheit und jedem Brückenmodul eine Maschinennummer zuweist und die Mitteilungsobjekte um die Maschinennummer erweitert.

13. Verfahren nach Anspruch 12, dadurch gekennzeichnet, daß man die Maschinennummer und ein Identifikationsfeld des Mitteilungsobjektes in einem inhaltsadressierbaren Speicher und weitere Felder in einem Freizugriffsspeicher speichert.

14. Verfahren nach Anspruch 13, dadurch gekennzeichnet, daß in den weiteren Datenfeldern der Mitteilungsobjekte Semaphore und Ereignis-Folge-Felder

enthalten sind, wobei das Ereignis-Folge-Feld auf eine Folge hinweist, die ausgelöst wird, wenn sich der Zustand der Semaphore ändert.

15. Verfahren nach Anspruch 14, dadurch gekennzeichnet, daß das Ereignis-Folge-Feld einen Verweis auf eine Speicherort in einem Ereignis-Folge-Speicher aufweist. 5

16. Datenverarbeitungsanlage mit mindestens zwei Datennetzwerken mit verteilten Rechnerknoten und mit einer Schnittstelle nach einem der Ansprüche 1 bis 10 11.

---

Hierzu 3 Seite(n) Zeichnungen

---

15

20

25

30

35

40

45

50

55

60

65

**BEST AVAILABLE COPY**

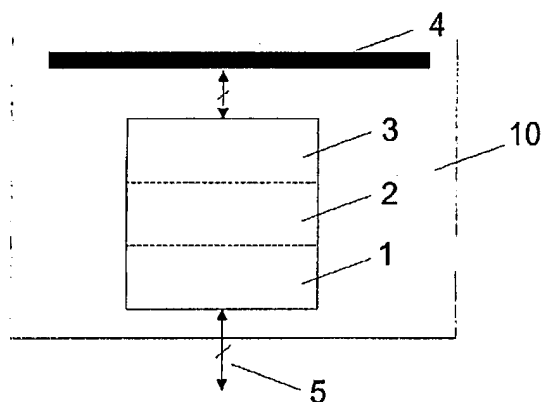


Fig. 1

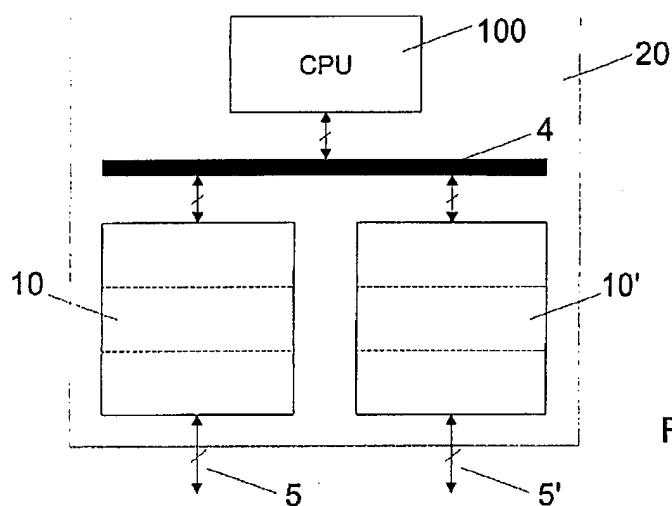


Fig. 2

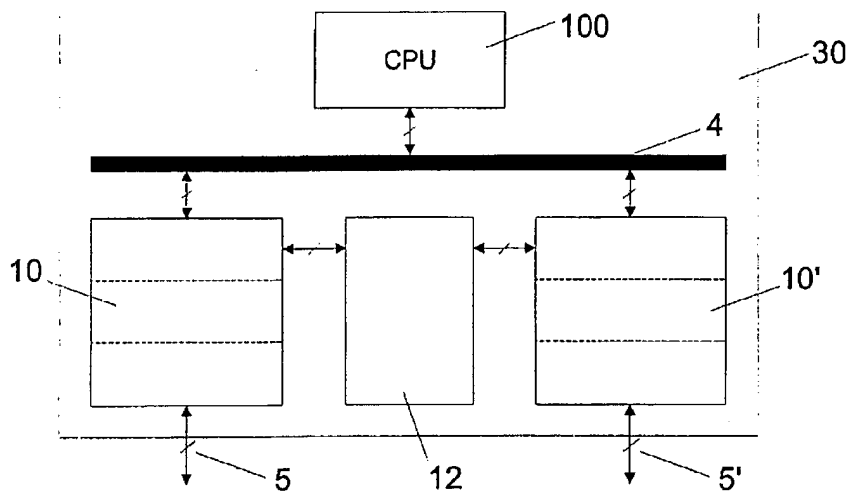


Fig. 3

BEST AVAILABLE COPY

902 026/471

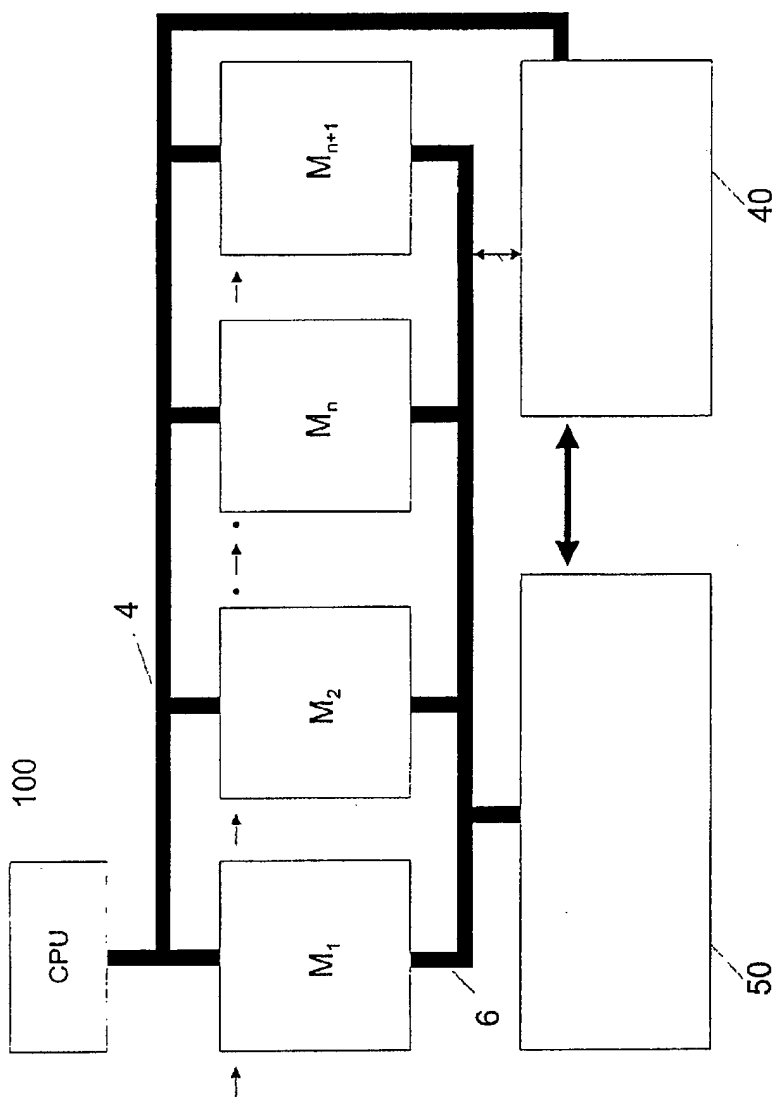


Fig. 4

BEST AVAILABLE COPY

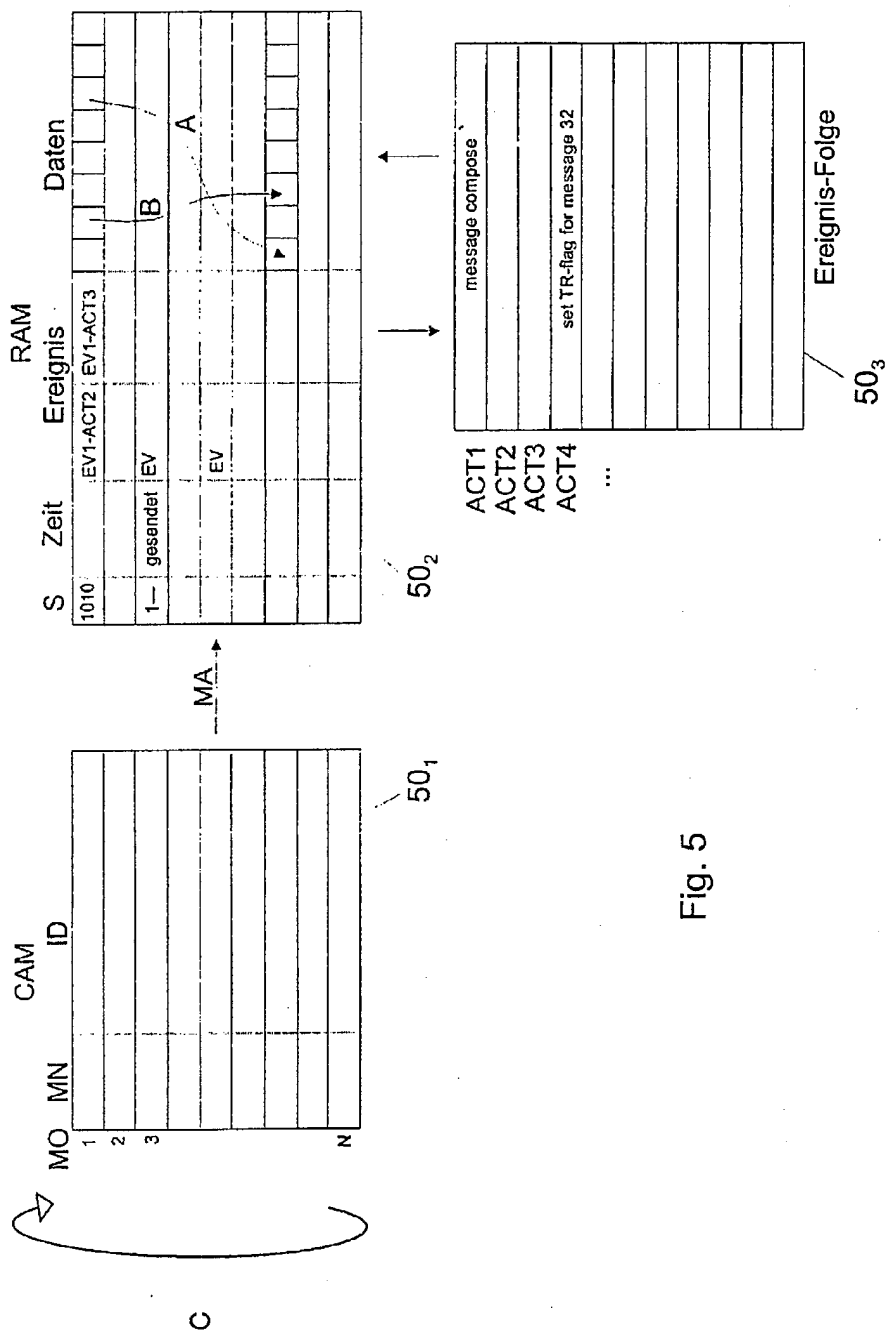


Fig. 5

BEST AVAILABLE COPY